Conceptual Clustering of Social Bookmarking Sites

Miranda Grahl, Andreas Hotho, Gerd Stumme
Knowledge & Data Engineering Group, University of Kassel, Germany,
http://www.kde.cs.uni-kassel.de, {lastname}@cs.uni-kassel.de

Research Center L3S, Hannover, Germany, http://www.l3s.de

Abstract: Currently, social bookmarking systems provide intuitive support for browsing locally their content. A global view is usually presented by the tag cloud of the system, but it does not allow a conceptual drill-down, e. g., along a conceptual hierarchy. In this paper, we present a clustering approach for computing such a conceptual hierarchy for a given folksonomy. The hierarchy is complemented with ranked lists of users and resources most related to each cluster. The rankings are computed using our FolkRank algorithm. We have evaluated our approach on large scale data from the del.icio.us bookmarking system.
Key Words: folksonomies, knowledge management, data mining
Category: H.3.7, H.5.4

1 Introduction

Social resource sharing systems are a way of collaboratively organising collections of resources, and are thus a promising alternative to classical knowledge management approaches. Recent applications of resource sharing systems address primarily private issues like photo collections. Their high acceptance on the Web shows their high social impact. By today, the economical impact is only visible on the horizon, but it indicates a large market. IBM, for instance, announces experiments with folksonomies in their intranet, because the currently used taxonomy is too expensive to be maintained [4]. Microsoft also intends to invest in this area [5].

The easy use of resource sharing systems makes them good candidates for knowledge management applications in a commercial setting, at least in domains where stronger structured approaches like ontologies could not take hold yet, or where their maintenance is too costly. This will hold especially in domains where people with no experience in data modelling have to deal with the tools.

The underlying structure of social resource sharing systems are so-called *folksonomies*, i. e., taxonomies created by the folk. A folksonomy consists of the *personomies* of its users. A personomy is the collection of all resources of a user, combined with a set of *tags*, which are catchwords that can be chosen arbitrarily by the users. Navigation in social resource sharing systems goes along hyperlinks which allow, for instance, to visit for a given tag, a web page listing all resources which have been tagged with this tag by at least one user. These systems allow thus for direct search of relevant entries. They also allow for serendipitous browsing, by following links to tags, users, and/or resources in a more or less random way. In a nutshell, folksonomy based systems are tuned for search and local navigation.

With their tag clouds, social resource sharing systems also provide a simple mean to discover the overall content of their folksonomy. However, when the set of all tags becomes too large, one is looking for more structured ways of presenting the folksonomy's content.

In this paper, we present a conceptual, hierarchical clustering approach for folksonomies, and discuss its value for analyzing the content of a folksonomy. First, we make iterative use of partitioning clustering (using two times the k-Means algorithm in our setting) on the set of tags. This step is followed by an application of our FolkRank algorithm [9] to discover resources and users that are related to the leaf clusters of the resulting cluster hierarchy, leading to the discovery of communities of interest.

The generation of the cluster hierarchy is completely automatic, and may serve as input for a manual creation of a concept hierarchy (ontology) in a subsequent step.

In order to evaluate our approach, we have analyzed the large-scale popular social bookmarking sytem del.icio.us.[1] Del.icio.us is a server-based system with a simple-to-use interface that allows users to organize and share bookmarks on the internet.

1.1 State of the Art

The most similar feature to our approach that is implemented in del.icio.us is its tag cloud http://del.icio.us/tag/ . The cloud is probably based on the frequency of tags only, no clustering is involved. It provides a first, global overview over the content of the system, but does not allow further conceptual drill-down into the data, since a click on one of the tags leads directly to an unstructured list of bookmarks.

Clustering of tags is one approach to support browsing and search within social bookmarking sites and is therefore of large interest. The scientific work that is most similar to ours is presented by Begelman et.al. in [1]. A graph based clustering approach called Metis is used on the weighted tag co-concurrence graph to find tag clusters in del.icio.us and RawSugar data.

Simpler approaches are using only the weights of the tag co-concurrence network to split the graph into independent subgraphs. Every subgraph is then considered as a cluster [10]. The exploration of the network along the co-concurrence graph is discussed in the blog of Rashmi Sihna.[2] Simple probabilistic methods or association rule mining approaches are used to extract relation between tags in [12] and [11]. A hierarchical clustering approach is applied on the weighted tag graph in [8] in order to compute a tag hierarchy. Similar clustering approaches are used to construct a hierarchy of tags for blogs in [3].

2 Dataset, Notations, and Algorithms

In the next subsection, we briefly present the dataset that we used for our experiment, and introduce some notations. Then we recall the basics of the clustering and the ranking algorithm that we used.

2.1 Dataset and Basic Notations

We have evaluated our approach on the social bookmarking sytem del.icio.us.[3] Between July 27 and 30, 2005, we crawled del.icio.us and obtained a set U of 75,085 users, a set

[1] http://del.icio.us

[2] http://www.rashmisinha.com/archives/05_02/tag-sorting.html

[3] http://del.icio.us

T of 456,666 tags, and a set R of 3,006,114 resources [9]. There were in total 7,281,940 posts, i. e., triples of the form (u, S, r), indicating that user $u \in U$ has assigned all tags contained in $S \subseteq T$ to resource $r \in R$. The set $Y \subseteq U \times T \times R$ of all tag assignments, i. e., of all $(\text{user}, \text{tag}, \text{resource})$ triples that show up in at least one post, consisted of 17,362,082 tag assignments.

2.2 *k*-Means – a Clustering Algorithm

For our experiments we used the well known cluster algorithm KMeans [7] as it provides in many cases good results. For k-Means, objects have to be represented in an n-dimensional vector space. As we will be working with a tag-tag-co-occurrence matrix (see Section 3.3 for details), our objects will be tags, as well as each dimension (feature) of the vector space.

The principle of KMeans is as follows: Let k be the number of desired clusters. The algorithm starts by choosing randomly k data points of D as starting centroids and assigning each data point to the closest centroid (with respect to the given similarity measure; in our case the cosine measure). Then it (re-)calculates all cluster centroids and repeats the assignment to the closest centroid until no reassignment is performed. The result is a non-overlapping partitioning of the whole dataset into k clusters.

Each cluster is described by its centroid. Usually one considers only the top n features of each centroid, ie those n dimensions of the vector space which have the highest values in the vector. To calculate the cluster description we extract the top n features from the centroid vector of the cluster P.

2.3 Folkrank – a Ranking Algorithm

To compute the users and resources that are most related to clusters of tags, we use the Folkrank approach (cf. [9]. Given a set of preferred tags, users, and/or resources of a folksonomy, Folkrank computes a topic specific ranking which provides an ordering of the elements of the folksonomy in descending importance with respect to the preferred elements.

Folkrank applies a two step approach to implement the weight-spreading ranking scheme on folksonomies. First, we transform the hypergraph between the sets of users, tags, and resources into an undirected, weighted, tripartite graph. On this graph A, we apply a version of PageRank [2] that takes into account the edge weights. The original PageRank algorithm computes $\mathbf{w} \leftarrow sA\mathbf{w} + (1 - s)\mathbf{p}$ iteratively until it converges or exceeds a given number of iterations. Here, A is the row stochastic version of the adjacency matrix of the graph, $\mathbf{p}$ is the preference vector (also called 'random surfer'), and $s \in [0, 1]$ is a damping factor. Our FolkRank algorithm computes a topic-specific ranking in a folksonomy by using a differential approach. It computes both a global ranking (i. e., PageRank with $s = 1$) and a ranking with a preference vector $\mathbf{p}$ which has increased entries for the topic at hand. The result is the difference between both rankings. Thus, we compute the winners and losers of the mutual reinforcement of resources when a user preference is given, compared to the baseline without a preference vector.

3 Constructing the Conceptual Hierarchy

In this section, we describe how our conceptual hierarchy is built. A global overview over the general idea is followed by a description of the technical details.

3.1 General overview

Social Bookmarking Systems suffer from their mass of information, which seems to be unstructured to the users. Especially beginners would like to get a fast and short overview about the tagged content. In a folksonomy, the content is described by (conjunctions of) tags. Since the set of all tags used is usually too large, we computed a three-level hierarchy of sets of tags (see Figure 1). The clusters on the lowest, most detailed level are complemented by lists of related resources and users (see Figure 1). The computation of this hierarchy is presented now. The resulting hierarchy is discussed in detail in Section 4.

3.2 Technical overview

For generating the conceptual hierarchy, we first removed, in a preprocessing step, some spam, and computed a vector space representation of the set of tags. Then we clustered the remaining set of tags, resulting in the lowest, most fine grained level of clusters. For each cluster, we extracted one tag as description. These descriptions were clustered again, yielding the middle layer of the conceptual hierarchy. Finally, we computed pairs of tags as descriptions of these 'meta-clusters', yielding the highest, most general level of the hierarchy.

These steps are described in detail in the remainder of this section, together with the computation of sets of related users and resources for each cluster on the most fine grained level.

Our proposed conceptual, hierarchical clustering approach is characterized by iterative application of the KMeans algorithm on a set of tags. The resulting leaf nodes of the hierarchy, we apply the FolkRank algorithm to facilitate easy access to relevant users, tags and resources.

3.3 Data Preprocessing

We started with the del.icio.us folksonomy as described in Section 2.1. First, we removed all posts with more than 50 tags, as they usually are spam. Then we computed, for each pair t_i, t_j of tags, their co-occurrence.

$$W(t_i,t_j) := |\{(u,r) \in U \times R \mid (u,t_i,r) \in Y \wedge (u,t_j,r) \in Y\}\,| \tag{1}$$

Each tag $t_i \in T$ is now represented in the reel $|T|$-dimensional vector space by the vector $\mathbf{t}^i := (\mathbf{t}^i_j)_{j=1,\ldots,|T|}$ with $\mathbf{t}^i_j := W(t_i,t_j)$ if $W(t_i,t_j) \geq 50$ and $i \neq j$, and $\mathbf{t}^i_j := 0$ else. We removed all tags that were represented by the $\mathbf{0}$ vector, as they are only peripheral to the folksonomy (see also [6]). The set T was thus reduced to 6356 tags. The remaining 'core' tags were then clustered.

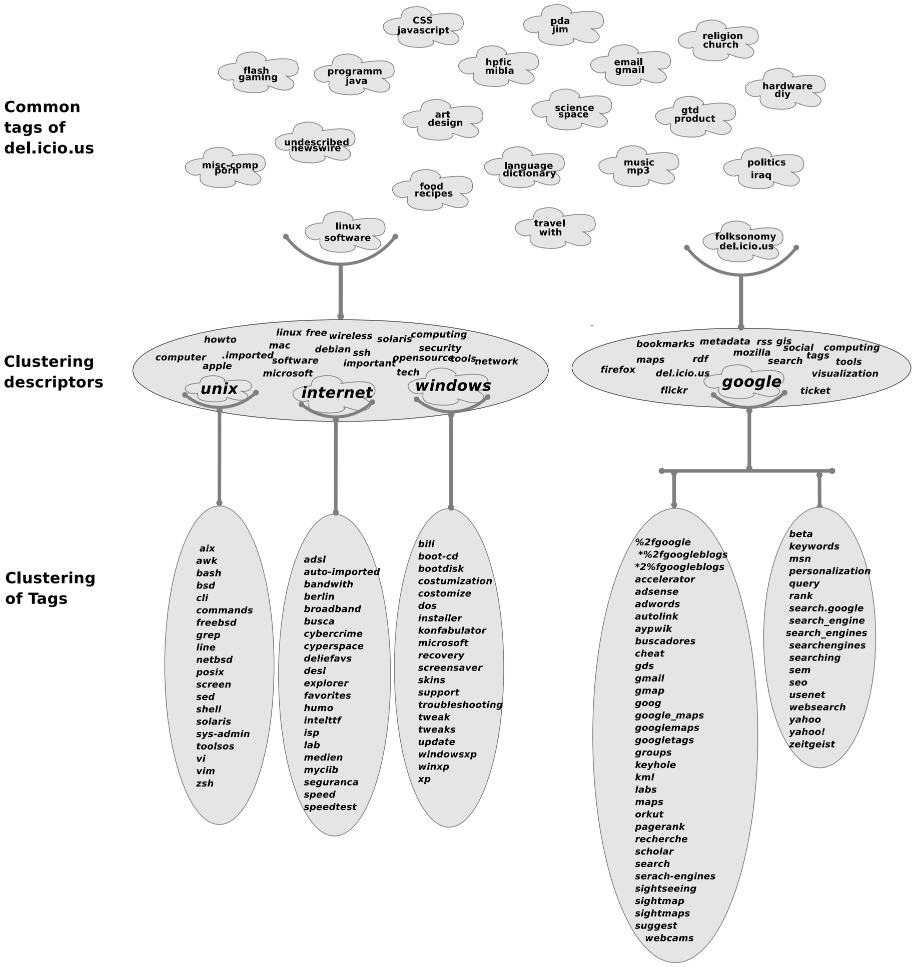

Figure 1: Conceptual hierarchy of the social bookmarking system del.icio.us.

3.4 Iterated Clustering

The conceptual hierarchy is computed as follows from bottom to top.

1. We cluster the set T of tags with k-Means with $k = 300$, resulting in a clustering $\mathbb{C} = \{C_1, \ldots, C_{300}\}$ where each cluster contains 21.18 tags in average. (Five of these clusters are displayed completely at the lowest layer of Figure 1.)

   For each cluster C_i, we extracted from its centroid the tag $\hat{t}_i$ with the hightest value as description of the cluster.[4] (The descriptors of 41 clusters are displayed in the middle layer of Figure 1.) We denote the set of all descriptors by $\hat{T} := \{\hat{t}_i | i = 1, \ldots, 300\}$. Please note that $|\hat{T}| = 274$ instead of 300, as one descriptor can result from more than one cluster (e. g., the tag 'google').

[4] Usually, one is taking more than one entry of the centroid as description, e. g., ten, but in our case, already the first tags turned out to be highly descriptive, contributing in average 67,41,% to the centroid.

2. While the set $\hat{T}$ provides already a good overview over the clusters computed above, it is still too large to be studied at a glance. Therefore, we clustered this set again with k-Means, this time with $k = 20$. We denote the result $\hat{\mathbb{C}} = \{\hat{C}_1, \ldots, \hat{C}_{20}\}$. (Two of the resulting clusters are displayed at the middle layer of Figure 1.) Again, we extracted for each cluster a description from its centroid. This time, however, the most central tag in the centroid is not significant enough, as it contributes in average only 14.45,% to the centroid. Therefore, we extracted the two most central tags from each centroid. (All 20 resulting tuples are shown in the top layer of Figure 1.) These tuples are a condensed summary of the current content of del.icio.us and enable the user to start a top-down navigation of the system.

3.5 Computing Related Users and Resources

After having structured the set of tags in a conceptual hierarchy, we complement now the most fine grained level of tag clusters with those users and resources which are most related to each cluster. I. e., for each of the clusters $C_1, \ldots, C_{300}$, we compute a ranking of users and resources, resp., according to their relevance to the tags contained in the cluster. To this end, we have applied, for each cluster C_i, the FolkRank with $s = 0.85$ and with a preference vector $\mathbf{p}^i$ composed as follows: $\mathbf{p}^i_j :=$ ... if tag $t_j \in C_i$ and ... else.

4 Results

We have applied the process presented in the previous section to the data of the del.icio.us system as described in Section 2.1.[5] Part of the resulting hierarchy is shown in Figure 1.

From top to bottom, the hierarchy allows us to explore the folksonomy in more and more detail. The top layer of the hierarchy is displayed completely in Figure 1. Its 20 pairs of tags provide a first overview over the content of del.icio.us (at the time of the crawl), and are thus comparable to the tag cloud at http://del.icio.us/tag/ . A main difference, however, is that the tag cloud of del.icio.us does not allow further drill-down along a conceptual hierarchy. When clicking on a tag in the tag cloud, del.icio.us directly presents all resources tagged with this tag. In our approach, we can navigate further down two more levels of the conceptual hierarchy.

For two selected entries of the top level, we have displayed the complete next level of the hierarchy. Let us first consider the subtree spanned by the tags 'linux' and 'software'. The next level shows a cluster consisting of 23 tags, including the tags 'linux', 'unix', 'apple', and 'windows. The fact that only 'linux' made its way further up to the top level (because it had the largest contribution to the centroid of the cluster) indicates the preference of the del.icio.us users concerning operating systems.

For three of the 23 tags, we have also displayed the full next (and last) level of the hierarchy. We see for instance that the tag 'unix' on the intermediate level summarizes many unix commands and tools, such as 'awk', 'bash', or 'vi'. The tag 'windows' on the

[5] We also applied the approach (with different parameters, due to the different sizes of the systems) to our BibSonomy system (http://www.bibsonomy.org). The results are not shown here due to space restrictions.

intermediate level represents a cluster which contains the microsoft operating systems 'xp' (in three variants) and 'dos', and issues like 'custonmization', 'installer', 'troubleshooting', or 'update'. The tag 'internet' on the intermediate level shows a wider variety of topics on the subsequent level, as might be expected.

The second branch of the hierarchy, spanned by folksonomy/del.icio.us, also provides some interesting insights. First, we observe that one tag in the subsequent layer is 'google', which, in contrast to the popularity of this search engine, did not make it to the top level of the hierarchy. This may again indicate a bias of the del.icio.us users. A second observation is more of a technical nature: We observe that 'folksonomy' is the most central component of the centroid of the rightmost cluster on the middle layer, even though the tag itself is not contained in the cluster itself. This effect results from the fact that we have set $\mathbf{t}_i^i = 0$ in the vector space representation. A third observation is that the tag 'google' in the intermediate layer leads to two different clusters in the fine grained layer. This results from the fact that both clusters displayed in the lower right of Figure 1 have a centroid in which 'google' is the largest entry. In fact, both clusters are related to Google, but address different aspects. The rightmost cluster is about search engines in general, including 'seo' [= search engine optimization], and the web applications Yahoo! and Zeitgeist. The second cluster from the right consists mainly of Google services, like Google Maps or Google Blog Search. This cluster contains also, to a lesser extent, research related tags: 'scholar', 'recherche', 'pagerank'.

Note that the displayed results are complete, and no cleaning-up has been performed. For a fully automatic approach, the resulting hierarchy is rather good. It might be used for a subsequent semi-manual generation of a taxonomy, in which eventually remaining inconsistencies are cleaned up.

If one is interested in the users or resources that are related to a cluster, one can use the results of PageRank. For the two Google clusters, the rankings are shown in Table 1. We see for instance, that the del.icio.us user 'ubi.quito.us' is the most relevant contributor to the Google service cluster, and the second most relevant contributor (behind user 'fritz') to the search engine cluster. The top URL in the service cluster is 'http://www.keyhole.com/kml/kml_tut.html', which refers to Google Earth, whereas the top URL in the other cluster, http://www.vaughns-1-pagers.com/internet/google-ranking-factors.htm, is about search engine optimization. Our conceptual hierarchy leads us thus also to (implicit) communities of interest among the del.icio.us users, and to clusters of related web pages.

5 Conclusion

In this paper, we have shown that the hierarchical application of partitioning clustering algorithms can provide a useful conceptual hierarchy on the set of tags, where the major components of the centroids of the clusters provide good descriptions. The leaves of the tag hierarchy have been extended with corresponding clusterings of the sets of resources and users, resp., to allow for accessing all dimensions of the folksonomy.

Acknowledgement. This work has partially been supported by the European Commission within the research project 'TAGora – Emergent Semantics in Online Social Communities'.

Table 1: Users and resources that are most related to the two Google clusters. The upper table relates to the left and the lower to the right cluster.

rank	user	rank	URL
0.002	ubi.quito.us	2.6E-4	http://www.keyhole.com/kml/kml_tut.html
0.002	kof2002	1.9E-4	http://www.googlesightseeing.com/
0.001	idealisms	1.9E-4	http://scholar.google.com/
6.4E-4	dajdump	1.9E-4	http://webaccelerator.google.com/
3.1E-4	dymphna	1.8E-4	http://www.shreddies.org/gmaps/
2.6E-4	laugharne	1.8E-4	http://www.arnebrachhold.de/2005/06/05/google-sitemaps-generator-v2-final
2.5E-4	konno	1.7E-4	http://www.google.com/webhp?complete=1&amp;hl=en
2.5E-4	preoccupations	1.6E-4	http://www.keyhole.com/kml/kml_doc.html
2.1E-4	josquin	1.5E-4	http://serversideguy.blogspot.com/2004/12/google-suggest-dissected.html
2.1E-4	wxpbofh	1.3E-4	http://www.google.com/help/cheatsheet.html

rank	user	rank	URL
0.001	fritz	1.5E-4	http://www.vaughns-1-pagers.com/internet/google-ranking-factors.htm
3.8E-4	ubi.quito.us	1.4E-4	http://www.google.com/press/zeitgeist.html
2.9E-4	kof2002	1.1E-4	http://www.philb.com/whichengine.htm
2.8E-4	triple_entendre	1.0E-4	http://inventory.overture.com/d/searchinventory/suggestion/
2.2E-4	cemper	9.9E-5	http://www.google.com/
1.7E-4	juanjoe	8.8E-5	http://www.buzzle.com/editorials/6-10-2005-71368.asp
1.5E-4	konno	7.8E-5	http://findory.com/
1.4E-4	tomohiromikami	7.4E-5	http://www.betanews.com/
1.2E-4	relephant	7.3E-5	http://clusty.com/
1.2E-4	masaka	7.1E-5	http://cgi.cse.unsw.edu.au/ collabrank/del.icio.us/

References

1. G. Begelman, P. Keller, and F. Smadja. Automated Tag Clustering: Improving search and exploration in the tag space. *WWW2006, May*, pages 22–26, 2006.
2. S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30(1–7):107–117, 1998.
3. C. H. Brooks and N. Montanez. Improved annotation of the blogosphere via autotagging and hierarchical clustering. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 625–632, New York, NY, USA, 2006. ACM Press.
4. Bud. Ibm's intranet and folksonomy. In *The Community Engine*. http://thecommunityengine.com/home/archives/2005/03/ibms_intranet_a.html, March 2005.
5. Bud. xfolk: An xhtml microformat for folksonomy. In *The Community Engine*. http://thecommunityengine.com/home/archives/2005/03/xfolk_an_xhtml.html, March 2005.
6. C. Cattuto, C. Schmitz, A. Baldassarri, V. D. P. Servedio, V. Loreto, A. Hotho, M. Grahl, and G. Stumme. Network properties of folksonomies. *AI Communications Special Issue on "Network Analysis in Natural Sciences and Engineering" (to appear)*, 2007.
7. E. Forgy. Cluster analysis of multivariate data: Efficiency versus interpretability of classification. *Biometrics*, 21(3):768–769, 1965.
8. P. Heymann and H. Garcia-Molina. Collaborative creation of communal hierarchical taxonomies in social tagging systems. Technical report, Stanford InfoLab, 2006.
9. A. Hotho, R. Jäschke, C. Schmitz, and G. Stumme. Information retrieval in folksonomies: Search and ranking. In Y. Sure and J. Domingue, editors, *The Semantic Web: Research and Applications*, volume 4011 of *LNAI*, pages 411–426, Heidelberg, June 2006. Springer.
10. P. Mika. Ontologies are us - a unified model of social networks and semantics. In *Proceedings of the 4th International Semantic Web Conference (ISWC)*, 2005.
11. C. Schmitz, A. Hotho, R. Jäschke, and G. Stumme. Mining association rules in folksonomies. In *Proc. IFCS 2006 Conference*, Ljubljana, July 2006.
12. P. Schmitz. Inducing ontology from flickr tags. In *Collaborative Web Tagging Workshop at WWW2006, Edinburgh, Scotland*, May 2006.